SIMUL.080C1 PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	David H. Jerome, et al.	Group Art Unit: Unknown
Appl. No.	:	Unknown	
Filed	:	Herewith	
For	:	METHOD AND SYSTEMS FOR A GRAPHICAL REAL TIME FLOW TASK SCHEDULER	
Examiner	:	Unknown At This Time	

PRELIMINARY AMENDMENT AND REQUEST FOR APPROVAL OF DRAWING CHANGES

Assistant Commissioner for Patents
Washington, D.C. 20231

Dear Sir:

Prior to examining the above-captioned continuation application filed herewith, please enter the following amendments. Applicants submit this Preliminary Amendment in the present continuation application in response to a Final Office Action mailed January 11, 2001, in connection with U.S. Patent Application No. 09/193,763, filed November 17, 1998, which is the parent application to the present continuation application.

IN THE DRAWINGS:

Applicants respectfully request approval of the following drawing changes shown in RED on the attached copies of the formal drawings as filed herewith. As shown in RED in Figure 11, Applicants request approval to delete the arrowhead pointing to state 1125 from the connecting line between state 1125 and state 1135, and Applicants request approval to correct the spelling of the word "INITILIZE" to "INITIALIZE" in state 1130.

IN THE SPECIFICATION:

Following the title, please add the following sentence as the first paragraph of the specification:

--This application is a continuation application under 35 U.S.C. § 120 of pending U.S. Patent Application No. 09/193,763, filed November 17, 1998, the entirety of which is hereby incorporated by reference.--

Please replace the paragraph beginning at page 6, line 16, with the following rewritten paragraph:

--In one embodiment, sequences constructed by the GUI client 205 are processed to a server 210 and saved in corresponding sequence databases 245. Thus, by way of example, each separate sequence created by the GUI client 205 has a separate sequence database 245. When a scheduler 225 signals a sequence execution process 220 to run a specific sequence, the sequence execution process 220 retrieves the sequence from the corresponding sequence database 245 to determine the steps to run. In one embodiment, the GUI client 205 exchanges information with the server 210, including for example, the data input by a user. The scheduler 225 has read access to the sequence database 215. When requested by the GUI client 205, the server 210 directs the scheduler 225 to put sequences on-line or take sequences off-line. When a sequence is on-line, it is ready to run under the direction of the scheduler 225. When the server 210 requests the scheduler 225 to place the sequence on-line, the scheduler 225 opens a sequence execution process 220 to control that particular sequence. Therefore, multiple sequence execution processes may be open at any given time.--

Please replace the paragraph beginning at page 7, line 6, with the following rewritten paragraph:

--The scheduler 225 maintains three collections of pointers to on-line sequences: on-line, scheduled, and running. The on-line collection may be sorted by name and contains pointers to the associated sequence execution process 220. The scheduled collection is sorted by next runtime and

contains those sequences that are on-line and scheduled. The running collection is sorted by next kill time and contains sequences that are on-line and running. Periodically, the scheduler 225 checks the scheduled and running collection for the next event. When the time occurs for a sequence to be started, the scheduler signals the appropriate sequence execution process 220 to run the sequence. The scheduler 225 monitors the sequence status in the sequence database 215 to determine when the sequence has completed execution.--

Please replace the four paragraphs beginning at page 7, line 25, with the following rewritten four paragraphs:

--In one embodiment, during execution of a sequence, the sequence execution process 220 may write messages to a message server 235. The message server 235 then writes the messages to a message log database 240 for access by the server 210. The sequence execution process 220 continues running the sequence while the messages are processed. To allow multiple sequence execution processes 220 to run at any given time, the message server 235 handles all messages from the sequence execution processes 220 on a first-come, first-serve basis.

Advantageously, when the GUI client 205 creates a sequence designated as a model sequence, the server 210 communicates with a model application server 250. The model application server 250 stores the model sequences in a model application database 255. The model application sequences are used to simulate real time sequences, and can be used to optimize the process. Details on optimization and modeling of material process systems are included in the co-filed applications entitled INTERACTIVE PROCESS MODELING SYSTEM and PROCESS MODEL GENERATION INDEPENDENT OF APPLICATION MODE both of which were filed on November 17, 1998 with Application Numbers 09/193,434 and 09/195,420, respectively, and which are hereby incorporated by reference in their entirety.

An example of the GUI interface according to one embodiment of the present invention is shown in Figure 3. For each sequence created, a sequence process flow diagram (PFD) window 300 is opened. In this embodiment of the PFD window, the name of the sequence is displayed in the identification bar 305. In one embodiment, each sequence has a unique identifying name. A series of pulldown menus 310 and a button bar 315 are provided for the user to interface with the sequence PFD window 300. The use of pulldown menus 310 and button bars 315 are well known in the art and therefore do not require further explanation herein. The sequence PFD window 300

10201.59723001

also includes a sequence display window 340 which provides a graphical display of the current sequence to the user. When the sequence PFD window 300 is first opened, the sequence display window 340 is blank. The user creates a sequence by using a keyboard, a mouse, or other pointing device such as a trackball or joystick to drop and drag tasks from the task palette 320 into the sequence display window 340. The tasks are selected from one of the several task palettes 320. To add the task to the current sequence, the user drags the task into the proper location in the sequence display window 340. The task palette 320 is subdivided into several categories. For example, in a general task window 325, basic tasks such as an input task, an output task and a custom task are displayed. Details of specific types of tasks are discussed below. In a model task window 335, tasks appear that would be used for a model application sequence. These are tasks such as load case, store case, and solve, which would not be used to control a plant, but would be used to simulate the control of a plant. The generic task window 330 displays tasks used during generic sequences. In one embodiment, the sequence display window 340 provides a continual visual display to the user of the tasks included in the current sequence.

An example of a process of creating and modifying a sequence using the sequence PFD window 300 is shown in Figures 4A-4D. In Figure 4A, an initial sequence is created which includes a start state 405 followed by a Task A 410 and a Task B 415. An exit branch 420 of Task B is a terminal exit branch which causes the sequence to stop running. In Figure 4B, the user has selected and dragged a third Task C 425 having two terminal exit branches 430 and 435 into the sequence display window 340. Task C 425 is selected from one of the task palettes 320. In Figure 4C, the initial sequence is connected to the new Task C 425 by a line 440. The line 440 is created by dragging the terminal exit branch 420 from Task B 415 onto Task C 425. By connecting Task C 425 to Task B 415, the terminal exit branch 420 of Task B 415 is deleted. At this point, the sequence has two terminal exit branches, 430 and 435, from Task C 425. In the present example, Task C 425 contains branching logic which, under certain conditions, would revert the user back to Task B 415. To establish this branch, the terminal point 435 from Task C 425 is connected back to the input of Task B 415 as shown in Figure 4D. This creates a recycle loop 445 in the sequence. Now the sequence in Figure 4D contains the third Task C 425 with one terminal exit branch 430. The details of each task can be displayed and modified by double-clicking on the respective task icon in the sequence display window 340 as discussed in detail below.--

10032786.102201

Please replace the paragraph beginning at page 10, line 18, with the following rewritten paragraph:

--Once all the tasks in a sequence have been fully specified, the sequence can be initialized. Clicking on the setup button on the button bar 315 of Figure 3 causes a dialog box 500 similar to that as shown in Figure 5 to be displayed on the screen. A title bar 501 of the dialog box 500 contains the name of the sequence. Below the title bar 501 are a series of tabs used to modify the sequence setup. The dialog box 500 shown when a general tab 502 is selected is shown in Figure 5. Selecting the general tab 502 displays a scheduling box 505. In the scheduling box 505, the user has the option of selecting whether a sequence is to be scheduled or not scheduled. If a sequence is scheduled, then the scheduling information for the sequence is used to determine when the sequence is executed. The scheduling information will be described below in further detail. If the sequence is not scheduled, then the sequence may be demand-activated by a task within another sequence. For example, a sequence for data reconciliation may not be scheduled but may be activated by another sequence or task which detects the process to be at steady state. A run information box 510 displays the current run number of the sequence. The run number is incremented automatically each time the sequence is executed. This number is used for creating unique objects and output file names for each run of the sequence. For example, data files on any sequence run may be saved using a filename which includes the run number. This ensures each data file has a separate name.--

Please replace the two paragraphs beginning at page 12, line 17, with the following rewritten two paragraphs:

--If the daily option is selected in the run occurs box 640 by clicking on the daily button 642, the daily option box 665 appears as shown in Figure 6B. The number of days between executions of the sequence may then be selected. For example, if one day is entered in a frequency box 670, the sequence would execute at the begin time 610 every day starting with the begin date 620. If the value entered in the frequency box 670 is greater than one, the sequence will be executed on the day designated as the begin date 620 and then every N days after that where N is the number of days entered.

Appl. No. : **Unknown**
Filed : **Herewith**

If the weekly option is selected in the run occurs box 640 by clicking on the weekly button 643, the weekly selection box 675 as shown in Figure 6C is displayed. How often on a weekly basis the sequence runs may then be selected using a weekly frequency box 680. For example, if one week is entered in the weekly frequency box 680, the sequence runs on the same days each single week. The days of the week on which the sequence run is to occur is selected using checkboxes 685. To run the sequence on a less frequent scale, the number in the weekly frequency box may be increased. This results in the sequence running only every N weeks on the specified days. The weekly selection box 675 also includes an indication of the next scheduled run time 660 based on the frequency selected.--

102201.58723001

Please replace the paragraph beginning at page 13, line 17, with the following rewritten paragraph:

--In addition to scheduling the sequences, each task in a sequence may be defined by double clicking on the selected task in the sequence display window 340, thereby displaying a task dialog box 700 for the task, one embodiment of which is shown in Figure 7. As described above, each sequence is composed of a list of tasks. Each task is associated with a separate task dialog box 700. In each task dialog box 700, the title bar 702 contains the name of the task. Below the title bar, general information about the task appears including the task type and task description. Under a general tab 705, the task can be designated as activate or inactivate by selecting an appropriate activation level, either active or inactive, from a status box 710. When inactive, the task is bypassed in the sequence and the next task following the “continue exit branch” of the inactive task is designated to be executed next.--

Please replace the paragraph beginning at page 14, line 4, with the following rewritten paragraph:

--The task dialog box 700 also contains an execution limit box 725 to limit the amount of time allowed to execute the task. The maximum amount of time for the task to execute can be entered in a limit time box 730. If a limit time is specified, an over-limit action should also be specified from the pick list 735. The over limit actions include, by way of example, logging an error message and continuing, aborting the task and continuing the next task, or stopping the entire

sequence. A notes tab 707 in the task dialog box 700 may be selected to allow the user to enter documentation associated with the particular task. These may include a short description of the task or detailed notes about the task.--

Please replace the paragraph beginning at page 15, line 7, with the following rewritten paragraph:

--An input task allows a user to import data from an external data source or from a file and to download the data to define an input and an output block. The imported data may contain plant history data to assist in optimizing the plant model. An output task is similar to the input task. However, in the output task, the user chooses export and upload options. The e-mail task sends an e-mail message to the designated address. The user enters the e-mail address in the text of the message, to send a message notifying an operator or other program of a critical failure or other designated message. The load case task is used to load a case or set of data into the flowsheet of the associated model application. The store case task allows the user to store the data currently loaded in the flowsheet. The solve task allows the user to solve a case, e.g., a simulation or optimization problem, that has been defined for the flowsheet.--

10037834.102201

Please replace the two paragraphs beginning at page 15, line 27, with the following rewritten two paragraphs:

--The data reconciliation pre-processing task performs the steps necessary to prepare a data reconciliation case for the flowsheet to be solved. The data reconciliation review task reviews the solution of a data reconciliation case and determines what task to perform next based upon the results. The optimization pre-processing task performs the pre-processing steps necessary to set up an optimization case for the flowsheet. The optimization review task reviews the results of an optimization solution. The implementation pre-processing task performs the pre-processing steps necessary before sending targets to the controllers. The model sequence activation control task controls the activation of on-line model sequences that are not scheduled. The activation is based on various criteria that is set up for each on-line model sequence.

A steady state detection task determines if the unit is steady or unsteady by monitoring the values of a set of process measurement points. Selecting a steady state detection task causes a

steady state detection task dialog box 1000 to be displayed as shown in Figure 10. The circumstances which define steady state can therefore be varied in each sequence. In the steady state detection task dialog box 1000, the number of periods to be monitored for steady state is set in a period monitoring selection box 1010. For example, a history of the last N (where N is the number of periods entered by the user) measured values of each point is obtained and a statistical test is performed to determine whether the value of the point has significantly changed. If the point has not significantly changed, the value is determined to be steady. The minimum percentage of individual points needed to be steady for the overall unit to be considered in steady state can be specified in a threshold box 1015. In a results box 1020, the average of all points "percent steady" values is displayed in a percent steady box 1025. The percent steady box 1025 value is compared with the minimum percent required for steadiness as entered in the threshold box 1015, and the final result is given to the user as steady or unsteady in a steadiness indication box 1030.--

Please replace the two paragraphs beginning at page 17, line 3, with the following rewritten two paragraphs:

--Proceeding to state 1125, the solution from state 1120 is checked to determine if the solution is valid. If the solution is valid, the sequence 1100 proceeds along the YES branch to state 1130. In state 1130, the sequence saves the solution and initializes the software for the next run. The sequence 1100 then proceeds to end state 1145.

Returning to state 1125, if the solution is not valid, the sequence 1100 proceeds along the NO branch to state 1135 to determine if more iterations are likely to produce a valid solution. If there is an indication additional iterations might produce a valid solution, the sequence proceeds along the YES branch back to state 1120. The sequence may remain in this loop until either the solution is valid or a determination is made more iterations are not likely to produce a valid solution.--

Appl. No. : **Unknown**
Filed : **Herewith**

IN THE CLAIMS:

Prior to an initial examination of this continuation application, please add Claims 44 and 45, please amend Claims 11, 13-15, 17, 18, 20, 24, 25, 28-35, 38, and 39, and please cancel Claims 1-10 and 40-43 after adding Claims 44 and 45, as follows:

11. (amended) A real time graphical task scheduler used to both optimize and control a material processing system comprising:

a graphical user interface;

a plurality of task icons capable of being displayed on the graphical user interface, wherein at least one of said task icons represents a task to be performed by the material processing system;

a sequence development window capable of being displayed on the graphical user interface, wherein at least two of the plurality of task icons are connected to define a sequence;

a sequence scheduler which controls the operation of the sequence;

an optimization modeler which calculates a plurality of input variables for the sequence to optimize the material processing system operation; and

a process controller receiving signals from the sequence and relaying said signals to the material processing system in real time.

13. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence is associated with a start time, a run-time, and an interval time.

14. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence scheduler activates the sequence in response to a start time.

15. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence scheduler deactivates the sequence in response to a passage of a run-time.

17. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence is designated to selectively optimize and control the material processing systems.

18. (amended) The real time graphical task scheduler of Claim 17, wherein the selectively optimized sequence is converted to a control sequence.

20. (amended) A method of creating a sequence of instructions for optimizing and controlling a material processing system, the method comprising the acts of:

selecting a first task from a list of task icons on a graphical user interface;

placing the first task in the sequence;

selecting a second task having a plurality of output branches from the list of task icons;

adding the second task to the sequence;

defining a relationship between the first task and the second task; and

selecting one of the plurality of output branches of the second task based upon a defined set of criteria, where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized.

24. (amended) The method of Claim 20, further comprising controlling at least a portion of the material processing system with the created sequence of instructions.

25. (amended) The method of Claim 20, further comprising simulating at least a portion of the material processing system with the created sequence of instructions.

28. (amended) A real time graphical task scheduler for optimizing and controlling a material processing system comprising:

a graphical user interface adapted to represent a plurality of sequences by associating a plurality of tasks in a specified relationship, where at least one of said tasks from the plurality of tasks is represented by a task icon; and

a process controller adapted to schedule and to enable and disable an activation state of the plurality of sequences.

29. (amended) The real time graphical task scheduler of Claim 28, wherein the at least one task icon in a sequence is color-coded to define the state of the corresponding task.

30. (amended) The real time graphical task scheduler of Claim 28, wherein an instance of a sequence from the plurality of sequences is associated with a start time, a run-time, and an interval time.

31. (amended) The real time graphical task scheduler of Claim 30, wherein the process controller activates the instance of the sequence at the start time.

32. (amended) The real time graphical task scheduler of Claim 31, wherein the process controller deactivates the instance of the sequence after expiration of the run-time.

33. (amended) The real time graphical task scheduler of Claim 30, wherein the process controller generates a next run-time based on the interval time.

34. (amended) The real time graphical task scheduler of Claim 28, wherein the sequence is designated to selectively optimize and control the material processing system.

35. (amended) A method of both optimizing and controlling a material processing system comprising:

defining a plurality of task sequences using a graphical user interface, where the plurality of task sequences model at least a portion of the material processing system;

scheduling operation parameters of the plurality of task sequences;

simulating operation of the material processing system with at least one of the plurality of task sequences modeled;

calculating at least one variable for operating the material processing system, where the calculating includes computations that are a result of the simulating operation of the material processing system; and

transferring the at least one variable to a process controller to vary the operation of the material processing system.

38. (amended) The method of Claim 37, wherein the at least one variable is calculated from the optimized mathematical model.

39. (amended) The method of Claim 37, wherein the at least one variable is an input variable.

44. (new) The real time graphical task scheduler of Claim 11, wherein the sequence scheduler deactivates the sequence in response to a detection of a steady state condition, which is unlikely to produce a valid solution.

45. (new) The method of Claim 20, wherein the defined set of criteria comprises a detection of a steady state condition, which is unlikely to produce a valid condition.

REMARKS

The present application is a continuation application of the parent application entitled "METHOD AND SYSTEMS FOR A GRAPHICAL REAL TIME FLOW TASK SCHEDULER," with Application Number 09/193,763, filed on November 17, 1998.

10207985.032701

Applicants respectfully request reconsideration of the rejections previously made by the Examiner in connection with the parent application in view of the foregoing amendments and the following comments. Marked-up versions of replacement paragraphs and rewritten claims are provided on separate pages starting with a page entitled "**VERSION WITH MARKINGS TO SHOW CHANGES MADE**," attached hereto following the signature page of this preliminary amendment. On the attached pages, insertions are shown as double underlined text, while deletions are placed in [brackets] and are in strikethrough text.

In a Final Office Action mailed January 11, 2001, the Examiner rejected Claims 11-39 of the parent application under 35 U.S.C. § 103(a). In response to the Final Office Action of January 11, 2001, Applicants cancelled Claims 11-39 in the parent application to speed issuance of the allowed claims. Applicants continue to pursue the cancelled claims from the parent application as well as new claims in the present continuation application. Claims 12, 16, 19, 21-23, 26, 27, 36, and 37 remain as originally filed in the parent application. Claims 11, 20, 24-25, 28-33, 35, 38, and 39 have been amended herein in accordance with the amendments to the claims filed by Applicants on October 30, 2000, in connection with the First Office Action Response of the parent case. Applicants have amended Claims 13-15, 17, 18, and 34 for the first time herewith and have written new Claims 44 and 45 to more distinctly and more completely claim Applicants' invention.

Amended Figure 11, submitted herewith, has been amended from Figure 11 originally filed in the parent application to correct a spelling error and a minor typographical error. No new matter has been added by this amendment.

The specification has been amended in this preliminary amendment to correct minor typographical and grammatical errors as will be described in further detail later.

Discussion of the Revisions to the Specification

Applicants have amended the specification of the present application as shown in the replacement paragraphs provided herein. No new matter has been introduced to the present application.

In particular, the paragraphs beginning at page 6, line 16 and page 7, line 6 have been amended to correct typographical errors. The four paragraphs beginning at page 7, line 25 have been amended to correct antecedent basis, to correct typographical errors, and to add application

numbers to previously cross-referenced applications. The paragraph beginning at page 10, line 18, the two paragraphs beginning at page 12, line 17, and the paragraph beginning at page 13, line 17 have been amended to correct typographical errors. The paragraph beginning at page 14, line 4 has been amended to correct antecedent basis and to correct a typographical error. The paragraph beginning at page 15, line 7 has been amended to correct typographical errors. The two paragraphs beginning at page 15, line 27, and the two paragraphs beginning at page 17, line 3 have been amended to correct a typographical and grammatical errors.

Discussion of the Rejection of Claim 11 Under 35 U.S.C. § 103(a)

Applicants respectfully traverse the Examiner's rejection of Claim 11. In the January 11, 2001 Final Office Action of the parent application, the Examiner rejected Claim 11 under 35 U.S.C. § 103(a) as being obvious over U.S. Patent No. 5,850,221 to Macrae, et al., ("Macrae") in view of "Software Simulation," Proceeding of the 1996 Winter Simulation Conference, December 8-11, 1996, by Jerry Banks ("Banks"). In the January 11, 2001 Final Office Action of the parent application, the Examiner stated that "*Applicant's arguments filed 10/30/2000 have been fully considered but some are not persuasive.*" As will be explained in the discussion below, the Examiner has mischaracterized the teachings of both Macrae and Banks and has not clearly communicated the basis for the Office's findings. As set forth in M.P.E.P. § 2144.08, "[o]ffice personnel should not evaluate rebuttal evidence for its "knockdown" value against the prima facie case, *Piasecki*, 745 F.2d at 1473, 223 USPQ at 788, or summarily dismiss it as not compelling or insufficient. If the evidence is deemed insufficient to rebut the prima facie case of obviousness, Office personnel should specifically set forth the facts and reasoning that justify this conclusion." As discussed below, Macrae is nonanalogous art, there is no suggestion or motivation to combine Macrae and Banks, a combination of Macrae and Banks fails to teach or suggest all the claim limitations set forth in Claim 11, and Banks is not enabled. Thus, Macrae and Banks fail to teach the claimed invention.

In the January 11, 2001 Final Office Action of the parent case, the Examiner stated that "*Applicant argues that material processing is not disclosed in the prior art of records. However, to begin with, the term material processing is a broad term encompassing any tangible product. Moreover, Macrae's invention is not limited to health care plan [sic]. Although Macrae is specifically related to implementing medical health care plans, but through modifications and*

10021785.102301

variations it will be apparent to those skilled in the art to apply Macrae's invention to visually or graphical [sic] implement another object or material such as in a plant factory."

Applicants maintain that Macrae is nonanalogous art. As set forth in M.P.E.P. 2141.01(a), "[i]n order to rely on a reference as a basis for rejection of an applicant's invention, the reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." *In re Oetiker*, 977 F.2d 1443, 1446, 24 U.S.P.Q.2d 1443, 1445 (Fed. Cir. 1992). The present invention is related to "methods and systems for a graphical task scheduler for a manufacturing plant." The Examiner stated that the term "material processing" is a "broad term," and has interpreted the term to encompass "any tangible product." However, Macrae fails to disclose material processing of any tangible product.

Applicants respectfully note that the performance of Macrae's invention does not result in "any tangible product." Rather, the "apparatus and method **generates** a plurality of **graphic images** representing a medical treatment plan," Macrae, abstract. Applicants respectfully maintain that the generation of graphical images on a computer screen is not the material processing of a tangible product as argued by the Examiner. Therefore, Applicants respectfully contend that Macrae is not "in the field of applicant's endeavor."

Applicants further maintain that Macrae is also not "reasonably pertinent to the particular problem with which the inventor was concerned." At a basic level, the problem with which Applicants are concerned is the automation of a material processing system. Applicants note that Macrae fails to disclose any interface between Macrae's apparatus or method and material processing equipment that would facilitate the automation of a material processing system. Therefore, Applicants assert that Macrae is not "reasonably pertinent" to the particular problem with which Applicants were concerned. Because Macrae is neither "in the field of applicant's endeavor," nor is "reasonably pertinent to the particular problem with which the inventor was concerned," Macrae is nonanalogous art, and Applicants respectfully submit that Macrae cannot be relied upon as a reference under 35 U.S.C. § 103.

Even assuming arguendo that Macrae is analogous art, Applicants respectfully submit that there is no suggestion or motivation in the prior art to combine the teachings of Macrae and Banks. As stated in M.P.E.P. § 2143.01, "[t]he mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the

desirability of the combination." *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). In addition, as stated in M.P.E.P. § 2144.08(II)(A), "[t]o establish a prima facie case of obviousness in a genus-species chemical composition situation, as in any other 35 U.S.C. 103 case, it is essential that Office personnel find some motivation or suggestion to make the claimed invention in light of the prior art teachings." See, e.g., *In re Brouwer*, 77 F.3d 422, 425, 37 USPQ2d 1663, 1666 (Fed. Cir. 1996).

Applicants note that the Examiner has not provided a suggestion or a motivation to one having ordinary skill to combine Macrae and Banks. On the contrary, in the January 11, 2001 Final Office Action of the parent case, the Examiner merely concluded that "[t]herefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to incorporate the efficient processing used in Banks, such as using arithmetic combinations (mathematical) in the simulation with Macrae." Since neither Macrae nor Banks provides a suggestion to combine the references, Applicants assert and further discuss below that it would not have been obvious for one having ordinary skill in the art at the time the invention was made to combine Macrae and Banks.

One of ordinary skill in the art at the time the invention was made would not have been motivated to combine Macrae and Banks because a combination of Macrae and Banks lacks utility. As described earlier, Macrae "generates a plurality of graphic images representing a medical treatment plan." Macrae results in graphic images. Macrae neither executes nor performs the actual medical treatment plan that is displayed. When Macrae discusses the term "simulating," it is only in the context of verifying the correctness of a template, "[t]he Flow Chart view is ideal for simulating plan delivery in order to ensure the correctness of a new template," see Col. 17, lines 42-43. Macrae defines a template as "a generic healthcare treatment plan, protocol, or guideline," see Col. 5, lines 5-6. When read in the context of the paragraph, it appears that the term "simulating" as used by Macrae means "displaying." Thus, the "simulating" in Macrae is not related to the simulation of a process under the control of Macrae, since as described earlier, Macrae neither controls nor automates any process, let alone a material processing system.

Therefore, a combination of Macrae and Banks is not possible. There is no process that is under the control of Macrae to which "the efficient processing used in Banks" as stated by the Examiner could apply. Thus, Applicants respectfully submit that a combination of Macrae and

10021596.102101

Banks would lack utility, and therefore, one of ordinary skill in the art would lack the motivation to make such a combination.

Applicants respectfully submit that even if Macrae and Banks were combined, the resulting combination of Banks and Macrae fails to teach or suggest the invention as claimed. As set forth in M.P.E.P. § 2143.03, in order to reject a claim under 35 U.S.C. § 103(a), "all the claim limitations must be taught or suggested by the prior art." *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). In addition, as also set forth in M.P.E.P. § 2143.03, "[a]ll words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 U.S.P.Q. 494, 496 (CCPA 1970).

The Examiner argued that "a real time graphical task scheduler used to both optimize and control a material processing system" is disclosed by Macrae. However, Applicants also note that, in discussing Claim 28, the Examiner acknowledged that "'a real time graphical task scheduler for optimizing and controlling material processing systems', is not explicitly disclosed" by Macrae. Nonetheless, in rejecting Claim 11, the Examiner cited the following, "*'a Scheduling and tasks management client software,' wherein the software 'uses the time slots to display tasks scheduled for individual users' (col. 27, lines 12-19) furthermore, 'the graphic images are presented in a chronological order based on real or visual time slots and may be viewed in either a flow chart or a chart view format' (Abstract),*" as support. Applicants respectfully traverse the Examiner's characterization of Macrae. Applicants note that the applicable standard recited in M.P.E.P. § 2143.03 requires that "[a]ll words in a claim must be considered in judging the patentability." For example, the following terms "real time," "task scheduler," "optimize," and "control," as recited and used by Applicants in Claim 11 are not described by Macrae.

Applicants use the term "real time" in Applicants' specification consistent with the ordinary meaning of the term. For example, "real time" is "the actual time during which something takes place," Merriam-Webster's Collegiate Dictionary (1997 10th ed.). Thus, the invention defined by Claim 11 optimizes and controls a material processing system while the material processing system is running. By contrast, Macrae states "[r]eal time is defined as the current active time of an order or plan," col. 8, lines 48-50. This means that "real time," as defined by Macrae, appears to be the mere storage of the present time, such as "11:00," rather than correspond to an activity that progresses as another takes place. See also the real time slot

object 363 of Figure 27 in Macrae. Macrae also defines "real time" in contrast to the definition provided by "virtual time," which Macrae defines as "a time set by a user," *id.* Applicants respectfully note that if Macrae had used "real time" in the ordinary meaning of the term, a more appropriate contrasting definition would have been "in a batch process" or the like. Applicants draw further support for Applicants' argument from the Abstract cited by the Examiner, where Macrae described presenting the time slots "in a chronological order based on real or virtual time slots." For Macrae to present multiple items in a graphical display from "real time" or "virtual time" in a chronological order, then there must be multiple times from which Macrae discerns the order. The presence of multiple times must imply that Macrae's "real time" is actually a history of time and does not relate to the performance of a task while Macrae's invention is operating.

Applicants respectfully note that the "tasks" that are scheduled by Macrae differ from tasks that are scheduled by Applicants' invention as defined by Claim 11. As cited by the Examiner, Macrae discloses "[t]he Scheduling and tasks management client software uses the time slots to display tasks scheduled for individual users," col. 27, lines 17-19. Clearly, these tasks that are scheduled in Macrae are tasks "for individual users" and are merely displayed to the user. Applicants' tasks are tasks that are "to be performed" in connection with a material processing system. To help clarify this distinction, Applicants have previously amended Claim 11 from "a task to be performed" to "a task to be performed by the material processing system."

With respect to the claim language "optimize and control a material processing system," a careful review of Macrae fails to disclose the optimization and control of a material processing system. Applicants respectfully request the Examiner to clarify where optimization and control are mentioned in Macrae. "All words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 U.S.P.Q. 494, 496 (CCPA 1970). Macrae does disclose a "processing means … for controlling the storage means," col 28, lines 22-24, but this "processing means" appears to relate to a computer processor (CPU), which does not process materials, and the storage means that is controlled appears to relate to a data store, such as a hard disk. Applicants respectfully request that the Examiner clearly articulate where optimization and control of a material processing system is disclosed by Macrae.

The Examiner further stated that Macrae discloses "processing means coupled to a display means," col. 28, line 23, "wherein the display includes, a plurality of graphic icon images," col. 28, lines 27-28, which the Examiner concluded discloses "a plurality of task icons

capable of being displayed on the graphical user interface, wherein at least one of said task icons represents a task to be performed by the material processing system." Applicants again assert that the "processing means" in Macrae refers not to a "material processing system," but rather to a computer processor, which does not process materials. Furthermore, Applicants respectfully note that the mere display of a task icon on a display screen does not disclose the control of a corresponding task performed by a material processing system.

The Examiner further stated that Macrae discloses "'a Scheduling and tasks management client software...' wherein, the software 'uses the time slots to display tasks scheduled for individual users.'" col. 27, lines 17-19, which the Examiner concluded discloses "a process controller receiving signals from the sequence and relaying said signals to the material processing system." As previously discussed, Macrae's "Scheduling and tasks management client software" does not disclose the control of a process. Rather, Macrae merely generates graphical images. Therefore, Macrae does not suggest or teach "relaying of said signals to the material processing system."

The Examiner further stated that Macrae discloses "an efficient execution, less time and reducing errors associated with processing the health care plan (col 1, line 41-col 2, line 20)." Applicants respectfully disagree and submit that the Examiner has mischaracterized Macrae. Macrae, from col. 1 line 41 to col. 2, line 20, describes the benefits and desirability of providing "an apparatus and method ... which will ... 1) reduce errors associated with communications between healthcare planners and providers ... and 4) copy and transfer medical treatment plans to various medical healthcare planner." The "processing" of the health care plan referred to by the Examiner does not relate to the execution of the health care plan by a user, but rather to the "reusing" of a "successful medical health treatment plan," see col. 2, lines 1-2. In describing the related art, Macrae provides "[f]or example, a leading cardiologist who develops a healthcare plan for a new surgical procedure may not be able to readily transfer the healthcare plan to other cardiologists to be adapted and used in their practices," see col. 2, lines 3-7. Thus, it appears to Applicants that Macrae's "efficient execution, less time and reducing errors" as characterized by the Examiner relate only to the copying or reusing a developed medical health treatment plan and not to an optimization or to a control of a material processing system. Thus, the Examiner's citation from Macrae is not relevant to the claimed invention.

The Examiner acknowledged that Macrae fails to disclose the invention as defined by Claim 11 because Macrae fails to disclose "an optimization modeler which calculates a plurality of input variables for the sequence to optimize the material processing system operation." The Examiner stated that Banks discloses "a simulation based scheduling software" and "an advance accomplishment of complex modeling tasks, significantly faster execution (optimization) using graphical user interface (Banks: pp. 31)." The Examiner concluded that "*[t]herefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to incorporate the efficient processing used in Banks, such as using arithmetic combinations (mathematical) in the simulation with Macrae.*" In the Examiner's "Response to Arguments" in the January 11, 2001 Final Office Action, the Examiner further stated that "*Applicant also argues that the prior art of records [sic] does not teach* ***optimization and controlling a material processing*** *system. However, this limitation is described in Banks. Bank [sic] describes speed of execution of the software, that is the speed and optimization of the application within a computer system which enables to accomplish complex modeling tasks.*"

Applicants respectfully submit that the subject matter that is optimized in the software described by Banks is completely different than the optimization and control of a material processing system as recited by Claim 11. The "optimization" referred to by Banks relates to the modeling software itself. Applicants reproduce the first four sentences of the paragraph from page 31 of Banks that were apparently relied upon by the Examiner in rejecting Claims 11 of the parent application.

Sentence 1:

"***GPSS/H is a product*** *of Wolverine Software Corporation (Crain and Smith 1995).*"

Sentence 2:

"***It is a*** *flexible, yet powerful* ***tool*** *for simulation.*"

Sentence 3:

"***It provides improvements*** *over GPSS V released many years earlier by IBM.*"

Sentence 4:
"***These enhancements*** *overcome the need to use external routines in FORTRAN to accomplish complex modeling tasks,* ***significantly faster execution****, an interactive debugging environment, a floating-point clock, built-in file I/O, use of named entity parameters, extended simulation control statements, built-in math and trig function, and ampervariables that allow complex arithmetic combinations to be used in the simulation.*"

From sentence 1, the subject of the paragraph is "GPSS/H," which is a software "product of Wolverine Software Corporation." In sentence 2, the subject of the sentence, "it," remains the GPSS/H software, and sentence 2 clearly indicates that GPSS/H is indeed software by describing "it" as a "tool for simulation." Sentence 3 further underscores that the subject of the paragraph, GPSS/H, is software. When sentence 3 describes "improvements," sentence 3 describes improvements to the GPSS/H software relative to an older version, "GPSS V," which was "released many years earlier by IBM." These "improvements" or "enhancements" are explained in greater detail in sentence 4. As the Examiner noted, these improvements relate to "significantly faster execution." Therefore, the "significantly faster execution" described by Banks relates only to the execution of the modeling software itself and not to a material processing system as recited in Claim 11.

By contrast, Applicants' invention as claimed optimizes a "material processing system," rather than the software itself. Clearly, the software itself and the "material processing system" that is optimized by the software are wholly separate entities. Therefore, even if it were proper to combine MacRae and Banks, Applicants respectfully submit that Applicants' invention as claimed is neither taught nor suggested by the combination thereof.

Applicants also note that the Examiner has not responded to Applicants' assertion that Banks is not enabled. Since Banks is not enabled, Banks should not be relied upon as a reference, see generally M.P.E.P. § 2121.01. For example, the Federal Circuit held that a brochure that merely boasted the ability and results of a claimed process "is insufficient, as a matter of law, to constitute an enabling disclosure of the process of the '903 patent," *Reading & Bates Construction Co. v. Baker Energy Resources Corp.*, 748 F.2d 645, 651, 223 U.S.P.Q. 1168 (Fed. Cir. 1984), see also M.P.E.P. § 2121.01. Similarly, Banks merely discloses brief product summaries of the abilities of a variety of software products. For example, the discussion of the GPSS/H software, on page 31, is about 20 lines long. Similarly, the discussions of the GPSS

World software and the SIMSCRIPT II.5 software, also appearing on page 31, are merely 17 lines and 20 lines long, respectively.

Applicants respectfully request that the Examiner reconsider the determination under 35 U.S.C. § 103 in accordance with the standard set forth in M.P.E.P. § 2144.08(III). M.P.E.P. § 2144.08(III) is reproduced below for the Examiner's convenience:

> "A determination under 35 U.S.C. 103 should rest on all the evidence and should not be influenced by any earlier conclusion. See, e.g., *Piasecki*, 745 F.2d at 1472-73, 223 USPQ at 788; *In re Eli Lilly & Co.*, 902 F.2d 943, 945, 14 USPQ2d 1741, 1743 (Fed. Cir. 1990). Thus, once the applicant has presented rebuttal evidence, Office personnel should reconsider any initial obviousness determination in view of the entire record. See, e.g., *Piasecki*, 745 F.2d at 1472, 223 USPQ at 788; *Eli Lilly*, 902 F.2d at 945, 14 USPQ2d at 1743. All the proposed rejections and their bases should be reviewed to confirm their correctness. Only then should any rejection be imposed in an Office action. The Office action should clearly communicate the Office's findings and conclusions, articulating how the conclusions are supported by the findings.
>
> Where applicable, the findings should clearly articulate which portions of the reference support any rejection. Explicit findings on motivation or suggestion to select the claimed invention should also be articulated in order to support a 35 U.S.C. 103 ground of rejection. *Dillon*, 919 F.2d at 693, 16 USPQ2d at 1901; *In re Mills*, 916 F.2d 680, 683, 16 USPQ2d 1430, 1433 (Fed. Cir. 1990). Conclusory statements of similarity or motivation, without any articulated rationale or evidentiary support, do not constitute sufficient factual findings."

Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, that a combination of Macrae and Banks fails to teach or suggest all the claim limitations set forth in amended Claim 11, and that Banks is not enabled. Applicants further respectfully submit that amended Claim 11 is patentably distinct over the cited art and respectfully traverses the Examiner's rejection of amended Claim 11.

Discussion of the Rejection of Claim 20 Under 35 U.S.C. § 103(a)

Applicants respectfully traverse the Examiner's rejection of Claim 20. In the January 11, 2001 Final Office Action of the parent application, the Examiner rejected Claim 20 under 35 U.S.C. § 103(a) as being obvious over Macrae in view of Banks. As described above with respect to Claim 11, Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, and that Banks is not enabled.

Applicants further respectfully submit that that a combination of Macrae and Banks, were it even proper to combine Macrae and Banks, still fails to teach or suggest all the claim limitations set forth in amended Claim 20.

In the January 11, 2001 Final Office Action, the Examiner stated that Macrae discloses "placing node in chronological order (col 8, lines 21-39)," which the Examiner cited as disclosing "placing the first task in the sequence." Applicants respectfully note that Macrae's placing of nodes in chronological order differs from Applicants' placing of a task in a sequence as claimed. As described earlier with respect to Claim 11, Macrae generates a visual or graphical display as an output. By contrast, "the sequence" as recited in amended Claim 20 refers not just to a graphical display, but also to an actual sequence of events in a material processing system. For example, amended Claim 20 recites "selecting one of the plurality of output branches of the second task based upon a defined set of criteria, where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized." As Applicants have previously discussed, Macrae fails to teach or suggest the control of a material processing system. Thus, Macrae does not disclose "placing the first task in the sequence."

The Examiner further stated that Macrae discloses "a process flow, wherein these connections indicate possible paths or branches a plan takes during delivery, and are illustrated as arrows, as seen in fig 19b (col. 12, lines 50-59)," which the Examiner argued discloses "selecting a second task having a plurality of output branches from the list of task icons, where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized." Applicants note that the Examiner has apparently misread this element, which is the third element of amended Claim 20. In a prior amendment filed in connection with the parent case, Applicants had previously amended the sixth element and not the third element of Claim 20 to introduce the "where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized" phrase to the claim. Applicants respectfully request the Examiner to provide further clarification regarding a rejection, if one remains, of Claim 20.

The Examiner stated that Macrae discloses "adding more order nodes by placing a node next to start node (Or1) will result the flow control to assign and display a second order node 2 (col 10, lines 39-58)," which the Examiner argued as disclosing "adding the second task to the sequence." As described earlier with respect to the first order node, Macrae merely generates a

visual or graphical display as an output. By contrast, "the sequence" as recited in Claim 20 refers not just to a graphical display, but also to an actual sequence of events in a material processing system. Since, Macrae fails to disclose the control of a material processing system, Macrae does not disclose "adding the second task to the sequence."

The Examiner further stated that Macrae discloses "the flow control node contains rules that select a branch at a decision point in a template or plan (col 7, lines 14-21)," which the Examiner concluded discloses "selecting one of the plurality of output branches of the second task based upon a defined set of criteria." Applicants note that the Examiner apparently has not read Claim 20 as amended. The element, as amended, reads "selecting one of the plurality of output branches of the second task based upon a defined set of criteria, where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized." Applicants respectfully submit that Macrae's flow control fails to teach or suggest selection of a task, because a task in Macrae is unrelated to a task that is performed by a material processing system. Further, Applicants respectfully note that Macrae fails to teach or suggest a simulation of a material processing system, and therefore, cannot use the results of a simulation to determine a criteria for selecting an output branch.

The Examiner acknowledged that Macrae fails to disclose "a method of creating a sequence of instructions for optimizing and controlling material processing systems." Rather, the Examiner stated that Banks discloses "a simulation based scheduling software, wherein, Banks further discloses an advance accomplishment of complex modeling tasks, significantly faster execution (optimization) using graphical user interface (Banks: pp. 31)." As explained in connection with Applicants' discussion of the Examiner's rejection of Claim 11, the "significantly faster execution" discussed in Banks refers only to the speed of the execution of the software, and not to the control or the optimization of a material processing system. Therefore, a combination of Macrae and Banks does not teach or suggest the invention as defined in amended Claim 20.

Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, that Banks is not enabled, and that a combination of Macrae and Banks, were it even proper to combine Macrae and Banks, still fails to teach or suggest all the claim limitations set forth in amended Claim 20. Therefore, Applicants respectfully maintain that Macrae, in combination with Banks, fails to teach or

10027985.102201

suggest Applicants' invention as defined by amended Claim 20, and Applicants respectfully traverse the Examiner's rejection of the claim.

Discussion of the Rejection of Claim 28 Under 35 U.S.C. § 103(a)

Applicants respectfully traverse the Examiner's rejection of Claim 28. In the January 11, 2001 Final Office Action of the parent application, the Examiner rejected Claim 28 under 35 U.S.C. § 103(a) as being obvious over Macrae in view Banks. As discussed above with respect to Claim 11, Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, and that Banks is not enabled. Applicants further respectfully submit that that a combination of Macrae and Banks, were it even proper to combine Macrae and Banks, still fails to teach or suggest all the claim limitations set forth in Claim 28.

In the January 11, 2001 Final Office Action, the Examiner stated that Macrae discloses "a 'Scheduling and tasks management client software,' wherein, the software 'uses the time slots to display tasks scheduled for individual users' (col 27, lines 12-19)," which the Examiner cited as disclosing "a real time graphical task scheduler for optimizing and controlling a material processing system." In an apparent contradiction, the Examiner proceeded to acknowledge that "'a real time graphical task scheduler for optimizing and controlling material processing systems,' is not explicitly disclosed" by Macrae.

Nonetheless, as Applicants have explained in greater detail earlier with respect to Claim 11, the Examiner has mischaracterized the teachings of Macrae. Macrae fails to disclose "a real time graphical task scheduler for optimizing and controlling a material processing system." Further, the tasks that are scheduled in Macrae are not related to either an optimizing or a controlling of a material processing system, let alone both optimizing and controlling a material processing system.

The Examiner stated that the same portion of Macrae, "a 'Scheduling and tasks management client software,' wherein, the software 'uses the time slots to display tasks scheduled for individual users' (col 27, lines 12-19)," discloses "a process controller adapted to schedule and enable and disable an activation state of the plurality of sequences." Applicants respectfully note that the cited portion of Macrae relates merely to a graphical display. As discussed with respect to Claim 11, Macrae does not disclose the control of a process and therefore, Macrae does not disclose

a process controller that is adapted to "schedule sequences and enable and disable an activation state of the plurality of sequences."

The Examiner acknowledged that Macrae fails to disclose "a real time graphical task scheduler for optimizing and controlling material processing systems." Rather, the Examiner stated that Banks discloses "a simulation based scheduling software, wherein, Banks further discloses an advance accomplishment of complex modeling tasks, significantly faster execution (optimization) using graphical user interface (Banks: pp. 31). As explained with respect to Claim 11, the "significantly faster execution" discussed in Banks refers only to the speed of the execution of the modeling software, and not to the control or to an optimization of a material processing system. Therefore, a combination of Macrae and Banks does not teach or suggest the invention as defined in amended Claim 28.

Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, that Banks is not enabled, and that a combination of Macrae and Banks, were it even proper to combine Macrae and Banks, still fails to teach or suggest all the claim limitations set forth in amended Claim 28. Therefore, Applicants respectfully maintain that Macrae, in combination with Banks, fails to teach or suggest Applicants' invention as defined by amended Claim 28, and Applicants respectfully traverse the Examiner's rejection of the claim.

Discussion of the Rejection of Claim 35 Under 35 U.S.C. § 103(a)

Applicants respectfully traverse the Examiner's rejection of Claim 35. In the January 11, 2001 Final Office Action of the parent application, the Examiner rejected Claim 35 under 35 U.S.C. § 103(a) as being obvious over Macrae in view of Banks. As described with respect to Claim 11, Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, and that Banks is not enabled. As discussed in greater detail below, Applicants further respectfully submit that a combination of Macrae and Banks, were it even proper to combine Macrae and Banks, still fails to teach or suggest all the claim limitations set forth in Claim 35.

In the January 11, 2001 Final Office Action, the Examiner stated that Macrae discloses "that a user defines each node and connects the nodes using a graphical user interface (col 5, lines 44-59, col 6, lines 55-67)," which the Examiner cited as disclosing "defining a plurality of task

sequences using a graphical user interface, where the plurality of task sequences model at least a portion of the material processing system." Applicants respectfully note that although a node in Macrae can be graphically connected to another node, the nodes do not correspond to models of items in a material processing system. Rather, the nodes are but mere "graphic elements or building blocks," see col. 6, lines 60-61. Macrae does not suggest that Macrae's nodes contain further information that would allow a node to maintain a model of an item in a material processing system. Thus, Macrae does not disclose "defining a plurality of task sequences using a graphical user interface, where the plurality of task sequences model at least a portion of the material processing system."

The Examiner further stated that Macrae discloses "a Scheduling and tasks management client software wherein, the software uses the time slots to display tasks scheduled for individual users (col 27, lines 12-19)" and further discloses "scheduling an integrated template or plan (col 26, lines 51-64) which the Examiner argued as disclosing "scheduling operation parameters of the plurality of task sequences." As described earlier in connection with Applicants' discussion of the rejection of Claim 11, Macrae's scheduled tasks are tasks "for individual users" and are merely displayed to the user. Moreover, Macrae fails to teach or suggest the setting of parameters during the operation of a material processing system because Macrae does not relate to a material processing system or to a control of a material processing system.

The Examiner further stated that Macrae discloses "the flow chart is ideal for simulating plan delivery in order to endure [sic] the correctness of a new template (col 17, lines 40-45)," which the Examiner argued as disclosing "simulating operation of the material processing system with at least one of the plurality of task sequences modeled." Applicants note that the quote from Macrae is actually "[t]he Flow Chart ***view*** is ideal for simulating plan delivery in order to ensure the correctness of a new template. The Chart view is more suited for delivering care to a real patient and more closely resembles patient charts." It appears to Applicants that Macrae's "Flow Chart" is, in fact, only a screen "view." It appears to Applicants that a "simulating" in Macrae is a mental step that is undertaken by the user when the user selects the "Flow Chart view" in order to proofread or check the chart that the user created. The Examiner incorrectly characterized this Flow Chart view as a simulation of the operation of the material processing system.

The Examiner further stated that Macrae, at col. 14, lines 30-51 discloses "calculating at least one variable for operating the material processing system, where the calculating includes

computations that are a result of the simulating operation of the material processing system." Macrae, at col. 14, lines 30-51, describes "orders which are repeated periodically, or run continuously, until certain criteria are met." The criteria include allowing a user to "further define the repeating order," to select "the date to start the next repetition of the order," to select "the maximum number of times the order is to be repeated," and the like. Applicants note that Macrae's criteria relate to the setting of a variable, as opposed to the calculation of a variable. It appears to Applicants that Macrae, at col. 14, lines 30-51 does not relate at all to a calculation of a variable, let alone a calculation of a variable that is the result of a simulation of a material processing system.

The Examiner further stated that Macrae discloses "transferring the assigned plurality of order icon images (col 3, lines 6-10)," which the Examiner argued as disclosing "transferring the at least one variable to a process controller to vary the operation of the material processing system." Applicants respectfully note that Macrae, at col. 3, lines 6-10, describes the transfer of "images" and not variables. Further, Macrae, at col. 3, lines 6-10, does not specify the location or the entity to which an image is transferred. Macrae fails to disclose the transfer of a "variable" to a "process controller." In fact, there is no description of a process controller in Macrae that is related to the operation of a material processing system. Thus, Applicants respectfully submit that Macrae fails to disclose "transferring the at least one variable to a process controller to vary the operation of the material processing system" as recited by amended Claim 35.

The Examiner acknowledged that Macrae fails to disclose "a method of both optimizing and controlling a material processing system." Rather, the Examiner stated that Banks discloses "a simulation based scheduling software, wherein, Banks further discloses an advance accomplishment of complex modeling tasks, significantly faster execution (optimization) using graphical user interface (Banks: pp. 31). As explained with respect to Claim 11, the "significantly faster execution" discussed in Banks refers only to the speed of the execution of the software, and not to the control or to the optimization of a material processing system. Therefore, a combination of Macrae and Banks does not teach or suggest the invention as defined in amended Claim 35.

Applicants respectfully submit that Macrae is nonanalogous art, that there is no suggestion or motivation to combine Macrae and Banks, that Banks is not enabled, and combination of Macrae and Banks fails to teach or suggest all the claim limitations set forth in amended Claim 35. Therefore, Applicants respectfully maintain that even if it were proper to combine Macrae and Banks, the combination of Macrae with Banks still fails to teach or suggest

10037869.102201

Applicants' invention as defined by amended Claim 35, and Applicants respectfully traverse the Examiner's rejection of the claim.

SUMMARY

In view of the foregoing amendments and discussions, Applicants respectfully submit that independent Claims 11, 20, 28, and 35, as amended, are patentably distinct over the cited art. Claims 12-19, 44, and 45 depend from Claim 11, further define Claim 11, and are likewise patentably distinct over the cited art. Claims 21-27 depend from Claim 20, further define Claim 20, and are likewise patentably distinct over the cited art. Claims 29-34 depend from Claim 28, further define Claim 28, and are likewise patentably distinct over the cited art. Claims 36-38 depend from Claim 35, further define Claim 35, and are likewise patentably distinct over the cited art. In view of the foregoing amendments and discussions, Applicants respectfully request the Examiner to withdraw the rejections to Claims 11-39 under 35 U.S.C. § 103(a) made by the Examiner in connection with the Final Office Action of the parent case.

Applicants further submit that that Claims 11-39, 44, and 45 and are in condition for allowance, and Applicants further request the Examiner to pass the present continuation application to the issue process. If there is any impediment to the allowance of the present continuation application, the Examiner is respectfully requested to call the undersigned attorney of record at (310) 407-3466 or at the number set forth below.

10027595 . 102201

Respectfully submitted,

KNOBBE, MARTENS, OLSON & BEAR, LLP

Dated: Oct. 22, 2001

By: Michael Okamoto

Michael S. Okamoto
Registration No. 47,831
Attorney of Record
620 Newport Center Drive
Sixteenth Floor
Newport Beach, CA 92660
(949) 760-0404

MSO-2333.DOC
100301

Appl. No. : **Unknown**
Filed : **Herewith**

VERSION WITH MARKINGS TO SHOW CHANGES MADE

Added material is double underlined.

Deleted material is bracketed and is in strikethrough text.

IN THE SPECIFICATION:

At page 6, the paragraph beginning at line 16:

--In one embodiment, sequences constructed by the GUI client 205 are processed to a server 210 and saved in corresponding sequence databases 245. Thus, by way of example, each separate sequence created by the GUI client 205 has a separate sequence database 245. When a scheduler 225 signals a sequence execution process 220 to run a specific sequence, the sequence execution process 220 retrieves the sequence from the corresponding sequence database 245 to determine the steps to run. In one embodiment, the GUI client 205 exchanges information with the server 210, including for example, the data input by a user. The scheduler 225 has read access to the sequence database 215. When requested by the GUI client 205, the server 210 directs the scheduler 225 to put sequences on-line or take sequences off-line. When a sequence is on-line, it is ready to run under the direction of the scheduler 225. When the server 210 requests the scheduler 225 to place the sequence on-line, the scheduler 225 opens a sequence execution process 220 to control that particular sequence. Therefore, multiple sequence execution [~~220~~] processes may be open at any given time.--

At page 7, the paragraph beginning at line 6:

--The scheduler 225 maintains three collections of pointers to on-line sequences: on-line, scheduled, and running. The on-line collection may be sorted by name and contains pointers to the associated sequence execution process 220. The scheduled collection is sorted by next runtime and contains those sequences that are on-line and scheduled. The running collection is sorted by next kill time and contains sequences that are on-line and running. Periodically, the scheduler 225 checks the scheduled and running collection for the next event. When the time occurs for a sequence to be started, the scheduler signals the appropriate sequence execution

process 220 to run the sequence. The [~~schedule~~] scheduler 225 monitors the sequence status in the sequence database 215 to determine when the sequence has completed execution.--

At page 7, the four paragraphs beginning at line 25:

--In one embodiment, during execution of a sequence, the sequence execution process 220 may write messages to [~~the~~] a message server 235. The message server 235 then writes the messages to a message log database 240 for access by the server 210. The sequence execution process 220 continues running the sequence while the messages are processed. To allow multiple sequence execution processes 220 to run at any given time, the message server 235 handles all messages from the sequence execution processes 220 on a first-come, first-serve basis.

Advantageously, when the GUI client 205 creates a sequence designated as a model sequence, the server 210 communicates with a model application server 250. The model application server 250 stores the model sequences in a model application database 255. The model application sequences are used to simulate real time sequences, and can be used to optimize the process. Details on optimization and modeling of material process systems are included in the co-filed applications entitled INTERACTIVE PROCESS MODELING SYSTEM and PROCESS MODEL GENERATION INDEPENDENT OF APPLICATION MODE [~~filed on even date herewith~~] both of which were filed on November 17, 1998 with Application Numbers 09/193,434 and 09/195,420, respectively, and which are hereby incorporated by reference in their entirety.

An example of the GUI interface [~~205~~] according to one embodiment of the present invention is shown in Figure 3. For each sequence created, a sequence process flow diagram (PFD) window 300 is opened. In this embodiment of the PFD window, the name of the sequence is displayed in the identification bar 305. In one embodiment, each sequence has a unique identifying name. A series of pulldown menus 310 and a button bar 315 are provided for the user to interface with the sequence PFD window 300. The use of pulldown menus 310 and button bars 315 are well known in the art and therefore do not require further explanation herein. The sequence PFD window 300 also includes a sequence display window 340 which provides a graphical display of the current sequence to the user. When the sequence PFD window 300 is first opened, the sequence display window 340 is blank. The user creates a sequence by using a

keyboard, a mouse, or other pointing device such as a trackball or joystick to drop and drag tasks from the task palette 320 into the sequence display window 340. The tasks are selected from one of the several task palettes 320. To add the task to the current sequence, the user drags the task into the proper location in the sequence display window 340. The task palette 320 is subdivided into several categories. For example, in a general task window 325, basic tasks such as an input task, an output task and a custom task are displayed. Details of specific types of tasks are discussed below. In a model task window 335, tasks appear that would be used for a model application sequence. These are tasks such as load case, store case, and solve, which would not be used to control a plant, but would be used to simulate the control of a plant. The generic task window 330 displays tasks used during generic sequences. In one embodiment, the sequence display window 340 provides a continual visual display to the user of the tasks included in the current sequence.

An example of a process of creating and modifying a sequence using the sequence PFD window 300 is shown in Figures 4A-4D. In Figure 4A, an initial sequence is created which includes a start state 405 followed by a Task A 410 and a Task B 415. An exit branch 420 of Task B is a terminal exit branch which causes the sequence to stop running. In Figure 4B, the user has selected and dragged a third Task C 425 having two terminal exit branches 430 and 435 into the sequence display window 340. Task C 425 is selected from one of the task palettes 320. In Figure 4C, the initial sequence is connected to the new Task C 425 by a line 440. The line 440 is created by dragging the terminal exit branch 420 from Task B 415 onto Task C 425. By connecting Task C 425 to Task B 415, the terminal exit branch 420 of Task B 415 is deleted. At this point, the sequence has two terminal exit branches, 430 and 435, from Task C 425. In the present example, Task C 425 contains branching logic which, under certain conditions, would revert the user back to Task B 415. To establish this branch, the terminal point 435 from Task C 425 is connected back to the input of Task B 415 as shown in Figure 4D. This creates a recycle loop 445 in the sequence. Now the sequence in Figure 4D contains the third Task C 425 with one terminal exit branch 430. The details of each task can be displayed and modified by double-clicking on the respective task icon in the sequence <u>display</u> window 340 as discussed in detail below.--

10023851.032402

At page 10, the paragraph beginning at line 18:

--Once all the tasks in a sequence have been fully specified, the sequence can be initialized. Clicking on the setup button on the ~~[toolbar]~~ button bar 315 of Figure 3 causes a dialog box 500 similar to that as shown in Figure 5 to be displayed on the screen. A title bar 501 of the dialog box 500 contains the name of the sequence. Below the title bar 501 are a series of tabs used to modify the sequence setup. The dialog box 500 shown when a general tab 502 is selected is shown in Figure 5. Selecting the general tab 502 displays a scheduling box 505. In the scheduling box 505, the user has the option of selecting whether a sequence is to be scheduled or not scheduled. If a sequence is scheduled, then the scheduling information for the sequence is used to determine when the sequence is executed. The scheduling information will be described below in further detail. If the sequence is not scheduled, then the sequence may be demand-activated by a task within another sequence. For example, a sequence for data reconciliation may not be scheduled but may be activated by another sequence or task which detects the process to be at steady state. A run information box 510 displays the current run number of the sequence. The run number is incremented automatically each time the sequence is executed. This number is used for creating unique objects and output file names for each run of the sequence. For example, data files on any sequence run may be saved using a filename which includes the run number. This ensures each data file has a separate name.--

At page 12, the two paragraphs beginning at line 17:

--If the daily option is selected in the run occurs box 640 by clicking on the daily button ~~[643]~~ 642, the daily option box 665 appears as shown in Figure 6B. The number of days between executions of the sequence may then be selected. For example, if one day is entered in a frequency box 670, the sequence would execute at the begin time ~~[650]~~ 610 every day starting with the begin date 620. If the value entered in the frequency box 670 is greater than one, the sequence will be executed on the day designated as the begin date 620 and then every N days after that where N is the number of days entered.

If the weekly option is selected in the run occurs box 640 by clicking on the weekly button 643, the weekly selection box 675 as shown in Figure 6C is displayed. How often on a weekly basis the sequence runs may then be selected using a weekly frequency box 680. For example, if one week is entered in the weekly frequency box 680, the sequence runs on the same

days each single week. The days of the week on which the sequence run is to occur is selected using checkboxes 685. To run the sequence on a less frequent scale, the number in the weekly frequency box may be increased. This results in the sequence running only every N weeks on the specified days. The weekly ~~[frequency]~~ selection box 675 also includes an indication of the next scheduled run time 660 based on the frequency selected.--

At page 13, the paragraph beginning at line 17:

--In addition to scheduling the sequences, each task in a sequence may be defined by double clicking on the selected task in the sequence display window 340, thereby displaying a task dialog box 700 for the task, one embodiment of which is shown in Figure 7. As described above, each sequence is composed of a list of tasks. Each task is associated with a separate task dialog box 700. In each task dialog box 700, the title bar 702 contains the name of the task. Below the title bar, general information about the task appears including the task type and task description. Under a general tab 705, the task can be designated as activate or inactivate by selecting an appropriate activation level, either active or inactive, from a status box 710. When inactive, the task is bypassed in the sequence and the next task following the "continue exit branch" of the inactive task is designated to be executed next.--

At page 14, the paragraph beginning at line 4:

--The task dialog box 700 also contains an execution limit box 725 to limit the amount of time allowed to execute the task. The maximum amount of time for the task to execute can be entered in ~~[the]~~ a limit time box 730. If a limit time ~~730~~ is specified, an over-limit action should also be specified from the pick list 735. The over limit actions include, by way of example, logging an error message and continuing, aborting the task and continuing the next task, or stopping the entire sequence. A notes tab 707 in the task dialog box 700 may be selected to allow the user to enter documentation associated with the particular task. These may include a short description of the task or detailed notes about the task.--

At page 15, the paragraph beginning at line 7:

--An input task allows a user to import data from an external data source or from a file and to download the data to define an input and an output block. The imported data may contain

plant history data to assist in optimizing the plant model. An output task is similar to the input task. However, in the output task, the user chooses export and upload options. The e-mail task sends an e-mail message to the designated address. The user enters the e-mail address in the text of the message, to send a message notifying an operator or other program of a critical failure or other designated message. The load case task is used to load a case or set of data into the ~~[flow sheet]~~ flowsheet of the associated model application. The store case task allows the user to store the data currently loaded in the ~~[flow sheet]~~ flowsheet. The solve task allows the user to solve a case, e.g., a simulation or optimization problem, that has been defined for the ~~[flow sheet.]~~flowsheet.--

At page 15, the two paragraphs beginning at line 27:

--The data reconciliation pre-processing task performs the steps necessary to prepare a data reconciliation case for the ~~[flow sheet]~~ flowsheet to be solved. The data reconciliation review task reviews the solution of a data reconciliation case and determines what task to perform next based upon the results. The optimization pre-processing task performs the pre-processing steps necessary to set up an optimization case for the flowsheet. The optimization review task reviews the results of an optimization solution. The implementation pre-processing task performs the pre-processing steps necessary before sending targets to the controllers. The model sequence activation control task controls the activation of on-line model sequences that are not scheduled. The activation is based on various criteria that is set up for each on-line model sequence.

A steady state detection task determines if the unit is steady or unsteady by monitoring the values of a set of process measurement points. Selecting a steady state detection task causes a steady state detection task dialog box 1000 to be displayed as shown in Figure 10. The circumstances which define steady state can therefore be varied in each sequence. In the steady state detection task dialog box 1000, the number of periods to be monitored for steady state is set in a period monitoring selection box 1010. For example, a history of the last N (where N is the number of periods entered by the user) ~~[measure]~~ measured values of each point is obtained and a statistical test is performed to determine whether the value of the point has significantly changed. If the point has not significantly changed, the value is determined to be steady. The

10037785.102201

minimum percentage of individual points needed to be steady for the overall unit to be considered in steady state can be specified in a threshold box 1015. In a results box 1020, the average of all points "percent steady" values is displayed in a percent steady box 1025. The percent steady box 1025 value is compared with the minimum percent required for steadiness as entered in the threshold box 1015, and the final result is given to the user as steady or unsteady in a steadiness indication box 1030.--

At page 17, the two paragraphs beginning at line 3:

--Proceeding to state ~~[1120]~~ 1125, the solution from state 1120 is checked to determine if the solution is valid. If the solution is valid, the sequence 1100 proceeds along the YES branch to state 1130. In state 1130, the sequence ~~[save]~~ saves the solution and initializes the software for the next run. The sequence 1100 then proceeds to end state 1145.

Returning to state 1125, if the solution is not valid, the sequence 1100 proceeds along the NO branch to state 1135 to determine if more iterations are likely to produce a valid solution. If there is an indication additional iterations might produce a valid solution, the sequence ~~[proceed]~~ proceeds along the YES branch back to state 1120. The sequence may remain in this loop until either the solution is valid or a determination is made more iterations are not likely to produce a valid solution.--

IN THE CLAIMS:

11. (amended) A real time graphical task scheduler used to both optimize and control a material processing system comprising:

a graphical user interface;

a plurality of task icons capable of being displayed on the graphical user interface, wherein ~~[each icon]~~ at least one of said task icons represents a task to be performed by the material processing system;

a sequence development window capable of being displayed on the graphical user interface, wherein at least two of the plurality of task icons are connected to define a sequence;

a sequence scheduler which controls the operation of the sequence;

an optimization modeler which calculates a plurality of input variables for the sequence to optimize the material processing system operation; and

a process controller receiving signals from the sequence and relaying said signals to the material processing system in real time.

13. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence ~~[has]~~ is associated with a start time, a run-time, and an interval time.

14. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence scheduler activates the sequence ~~[at]~~ in response to a start time.

15. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence scheduler deactivates the sequence ~~[after expiration]~~ in response to a passage of a run-time.

17. (amended) The real time graphical task scheduler of Claim 11, wherein the sequence is designated to ~~[either]~~ selectively optimize ~~[or]~~ and control the material processing systems.

18. (amended) The real time graphical task scheduler of Claim 17, wherein ~~[an]~~ the selectively optimized sequence ~~[may be]~~ is converted to a control sequence.

19. The real time graphical task scheduler of Claim 11, wherein the optimization modeler estimates appropriate input variables from history data of system operation.

20. (amended) A method of creating a sequence of instructions for optimizing and controlling a material processing ~~[systems]~~ system, the method comprising the acts of:

selecting a first task from a list of task icons on a graphical user interface;

placing the first task in the sequence;

selecting a second task having a plurality of output branches from the list of task icons;

adding the second task to the sequence;

defining a relationship between the first task and the second task; and

selecting one of the plurality of output branches of the second task based upon a defined set of criteria, where the defined set of criteria includes results of simulation of the material processing system such that the sequence is optimized.

24. (amended) The method of Claim 20, ~~[wherein the sequence of instructions is used to control the operation]~~ further comprising controlling at least a portion of the material processing system with the created sequence of instructions.

25. (amended) ~~[25.]~~ The method of Claim 20, ~~[wherein the sequence of instructions is used to simulate the operation]~~ further comprising simulating at least a portion of the material processing system with the created sequence of instructions.

28. (amended) A real time graphical task scheduler for optimizing and controlling a material processing ~~[systems]~~ system comprising:

~~[means for graphically creating]~~ a graphical user interface adapted to represent a plurality of sequences by ~~[combining]~~ associating a plurality of tasks in a specified relationship~~[; and]~~, where at least one of said tasks from the plurality of tasks is represented by a task icon; and

~~[means for controlling the scheduling and operation]~~ a process controller adapted to schedule and to enable and disable an activation state of the plurality of sequences.

29. (amended) The real time graphical task scheduler of Claim 28, wherein ~~[each of said]~~ the at least one task ~~[icons]~~ icon in a sequence is color-coded to define the state of ~~[said]~~ the corresponding task.

30. (amended) The real time graphical task scheduler of Claim 28, wherein ~~[the sequence has]~~ an instance of a sequence from the plurality of sequences is associated with a start time, a run-time, and an interval time.

31. (amended) The real time graphical task scheduler of Claim 30, wherein the ~~[means for controlling]~~ process controller activates the instance of the sequence at the start time.

32. (amended) The real time graphical task scheduler of Claim 31, wherein the ~~[means for controlling]~~ process controller deactivates the instance of the sequence after expiration of the run-time.

33. (amended) The real time graphical task scheduler of Claim ~~[32,]~~ 30, wherein the ~~[means for controlling]~~ process controller generates a next run-time based on the interval time.

34. (amended) The real time graphical task scheduler of Claim 28, wherein the sequence is designated to ~~[either]~~ selectively optimize ~~[or]~~ and control the material processing system.

35. (amended) A method of both optimizing and controlling a material processing system comprising:

defining a plurality of task sequences using a graphical user interface, where the plurality of task sequences model at least a portion of the material processing system;

scheduling ~~[the]~~ operation parameters of the plurality of task sequences;

simulating operation of the material processing system ~~[to calculate]~~ with at least one of the plurality of task sequences modeled;

calculating at least one variable for operating the material processing system, where the calculating includes computations that are a result of the simulating operation of the material processing system; and

transferring the at least one variable to a process controller to ~~[control]~~ vary the operation of the material processing system.

38. (amended) The method of Claim 37, wherein the at least one ~~[input]~~ variable is calculated from the optimized mathematical model.

39. (amended) The method of Claim 37, wherein the at least one ~~[input]~~ variable is an input variable.

10037386.102201

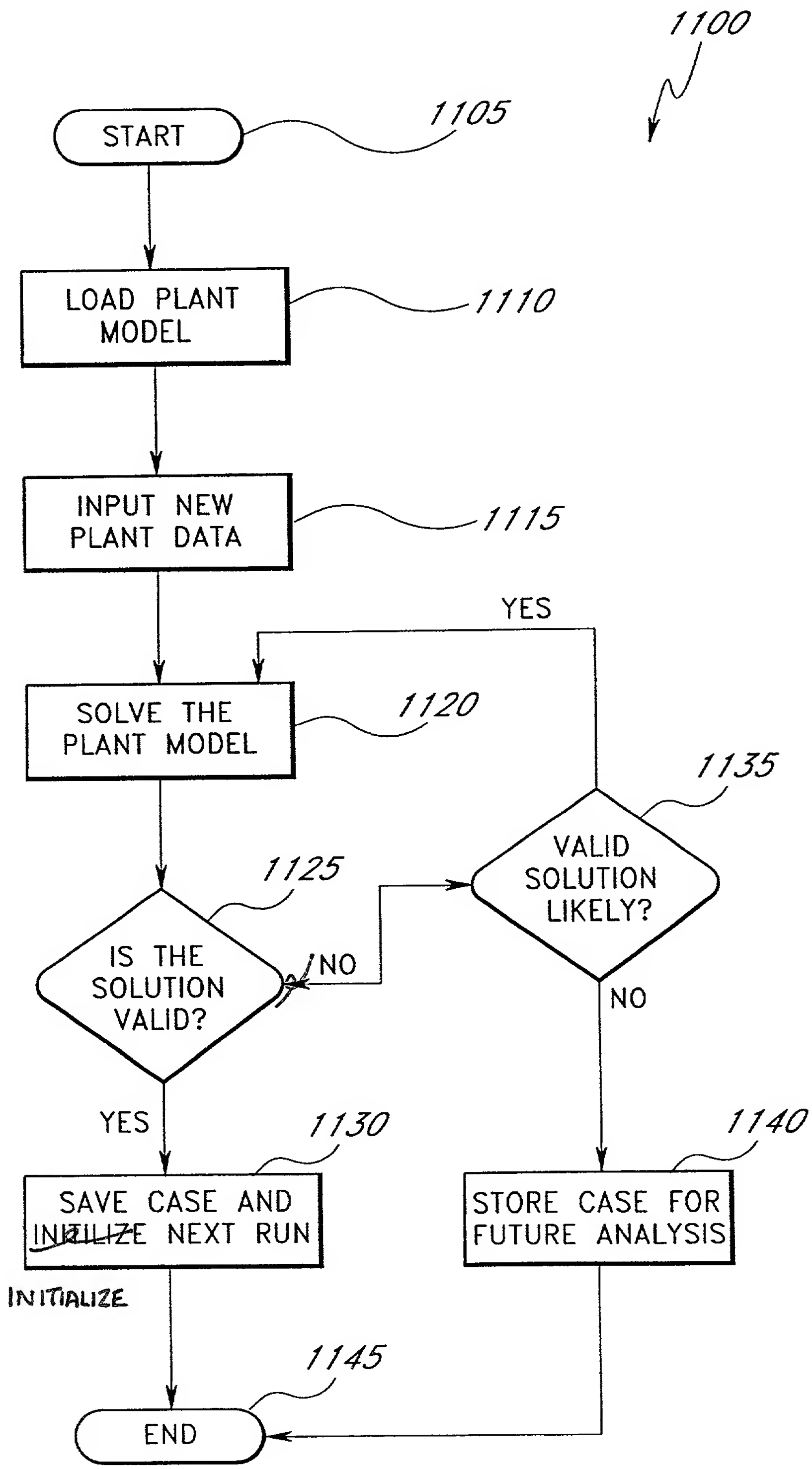

FIG. 11